Google™ Advanced Search Preferences Language Tools Search Tips

speculation mechanism Google Search

Web · Images · Groups · Directory · News ·

Searched the web for **speculation mechanism** . Results **21** - **30** of about **217,000**. Search took **0.22** seconds.

[PDF] 1 **SPECULATION** MAKES INFLATION WORSE There was much public concern ...
File Format: PDF/Adobe Acrobat - View as HTML
... However, **speculation** and wrong expectation will not be able to deviate inflation from ... and rising uncertainties in the world economy, market **mechanism** may not ...
www.mier.org.my/mierscope/drazmi25_11_2000.pdf - Similar pages

Code Reordering and **Speculation** Support for Dynamic Optimization ...
... This paper presents an approach called Precise **Speculation** that addresses these problems. The proposed **mechanism** is a component of our vision for Run-time ...
csdl.computer.org/comp/proceedings/ pact/2001/1363/00/13630163abs.htm - 11k - Cached - Similar pages
[More results from csdl.computer.org]

Speculation with BSA
... Figure 1 b) shows how a speculative control **mechanism**, such as the one in the ... **Speculation** is used when determining which block to next fire in the chain. ...
www.cs.waikato.ac.nz/timewarp/wengine/ papers/acac98/node3.html - 6k - Cached - Similar pages

[PPT] MASE Tutorial
File Format: Microsoft Powerpoint 97 - View as HTML
... work). Arbitrary **speculation mechanism** tested with blind load **speculation** study. Implementation was straight-forward in MASE. Checker ...
www.eecs.umich.edu/~larsone/mase.ppt - Similar pages

The Constraint Contextual Rewriting Project: Publications
... Abstract. In this paper, we propose a generic **mechanism** for extending decision procedures by means of a lemma **speculation mechanism**. ...
www.mrg.dist.unige.it/ccr/publications.html - 17k - Cached - Similar pages

sci.nanotech archives: **SPECULATION** AND QUESTIONS
... review of Moravec"; Next in thread: Alan Lovejoy: "Re: **SPECULATION** AND QUESTIONS". ... The **mechanism** proposed is "selective stickiness", but this does'nt sound any ...
discuss.foresight.org/critmail/ sci_nano.88-94/0564.html - 6k - Cached - Similar pages

[PS] The Performance Potential of Data Dependence **Speculation** & ...
File Format: Adobe PostScript - View as Text
... When the correct address becomes available and a misprediction. 1Note that the load **speculation mechanism** uses a subset of the mechanisms presented in [5]. ...
www2.cs.ucy.ac.cy/~yanos/papers/micro29.ps - Similar pages

[PDF] Observation of the CIV effect in interstellar clouds: a ...
File Format: PDF/Adobe Acrobat - View as HTML
... 6, DECEMBER 2000 Observation of the CIV Effect in Interstellar Clouds: A **Speculation** on the Physical **Mechanism** for Their Existence Anthony L. Peratt, Fellow ...
public.lanl.gov/alp/plasma/Perattpdf/ PerattVerschuurDec2000.pdf - Similar pages

Mechanisms of Action of Conjugated Linoleic Acid: Evidence and ...
... Mechanisms of Action of Conjugated Linoleic Acid: Evidence and **Speculation**. ... activities for CLA that beg the question of underlying biochemical **mechanism**. ...

12-9-03

h g gec e ch ec ech h e e e

www.blackwell-synergy.com/links/doi/10.1046/ j.1525-1373.2000.22302.x/abs/ - Similar pages

The **Mechanism** of Plate Tectonics
... Introduction; **Speculation**: There appears to be no agreed upon explanation of the **mechanism** that drives plate tectonics. (The notions ...
webspinners.com/dlblanc/tectonic/mechansm.html - 24k - Cached - Similar pages

◀ Goooooooooooogle ▶
Result Page: **Previous** 1 2 3 4 5 6 7 8 9 10 11 12 **Next**

speculation mechanism Google Search Search within results

Google Home - Advertise with Us - Business Solutions - Services & Tools - Jobs, Press, & Help

h g g e c e ch ec ech h e e e

Integrated Predicat d and Speculative Execution in the IMPACT EPIC Architecture (1998) (Make Corrections) (24 citations)

David I. August, Daniel A. Connors, Scott A. Mahlke, John W. Sias, Kevin M. Crozier, Ben-Chung Cheng, Patrick R. Eaton, Qudus B. Olaniran, Wen-mei W. Hwu

Proceedings of the 25th annual international symposium on Computer architecture

Abstract: Explicitly Parallel Instruction Computing (EPIC) architectures require the compiler to express program instruction level parallelism directly to the hardware. EPIC techniques which enable the compiler to represent control speculation, data dependence speculation, **and** predication have individually been shown to be very effective. However, these techniques have not been studied in combination with each other. This paper presents the IMPACT EPIC Architecture to address the issues involved in... (Update)

Context of citations to this paper: More

.... **with the question, Is general percolation really optimal A study combining sentinel scheduling with predicated execution for C programs [2] reported that for their benchmarks, 31 of cache misses and 13 TLB misses and page faults were spurious and thus could potentially be**...

.... **basic blocks from various control flow paths into one region to improve compiler optimization opportunities and scheduling [12] 4] [3].** These hyperblocks are typically formed from an inner most loop body. Basic blocks are incorporated into a region based on a heuristic...

Cited by: More
Chen-Yong Cher **and** T. N. Vijaykumar - School Of Electrical (Correct)
Compiling Application-Specific Hardware - Budiu, Goldstein (Correct)
Pegasus: An Efficient Intermediate Representation - Budiu, Goldstein (2002) (Correct)

Active bibliography (related documents): More All
0.4: Compiler Technology for Future Microprocessors - Hwu, Hank, Gallagher, Mahlke, .. (1995) (Correct)
0.4: Path Analysis **and** Renaming for **Predicated** Instruction .. - Carter, Simon.. (2000) (Correct)
0.4: **Predicated** Static Single Assignment - Carter, Simon, Calder, Carter.. (1999) (Correct)

Similar documents based on text: More All
1.0: Incorporating Predicate Information Into Branch Predictors - Simon, Calder, Ferrante (Correct)
0.5: Enhancing Loop Buffering of Media **and** Telecommunications.. - Sias, Hunter, Hwu (2001) (Correct)
0.5: Program Decision Logic Optimization Using Predication **And**.. - Hwu, August, Sias (2001) (Correct)

Related documents from co-citation: More All
10: Effective compiler support for **predicated execution** using the hyperblock - Mahlke - 1992
8: MediaBench: A Tool for Evaluating **and** Synthesizing Multimedia **and** Communications.. - Lee - 1997
8: A Framework for Balancing Control Flow **and** Predication - August, Hwu et al. - 1997

BibTeX entry: (Update)

D. I. August, D. A. Connors, S. A. Mahlke, J. W. Sias, K. M. Crozier, B. Cheng, P. R. Eaton, Q. B. Olaniran, **and** W. W. Hwu, "**Integrated Predicated and Speculative Execution** in the IMPACT EPIC Architecture," Proceedings of the 25th International Symposium on Computer Architecture, July 1998.
http://citeseer.nj.nec.com/august98integrated.html More

```
@inproceedings{ august98integrated,
```

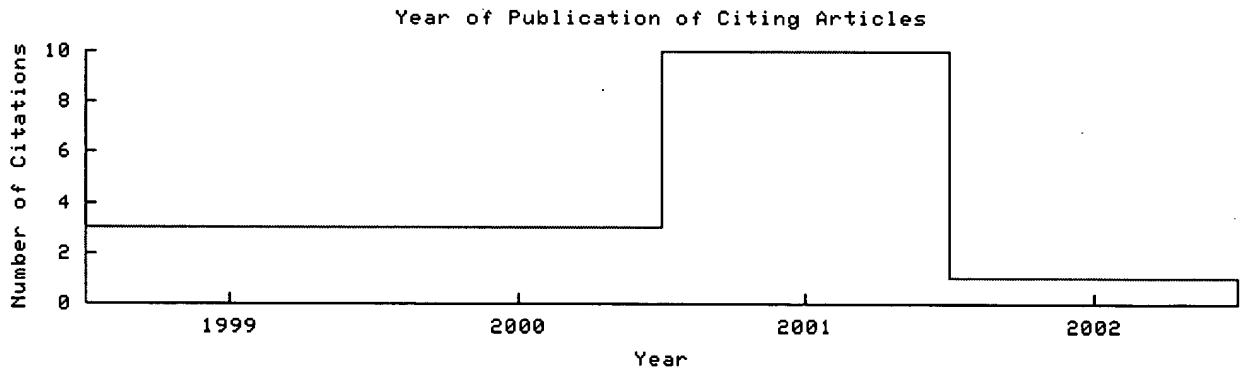
Retrieved 12-5-03

h c e ee ec c g eg e h

```
author = "David I. August and Daniel A. Connors and Scott A. Mahlke and John W.
title = "Integrated predicated and speculative execution in the {IMPACT {EPIC}}
booktitle = "Proceedings of the 25th annual international symposium on Computer
publisher = "IEEE Computer Society Press",
address = "1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA",
year = "1998",
url = "citeseer.nj.nec.com/august98integrated.html" }
```

Citations (may not include all citations):

126 Conversion of control dependence to data dependence (context) - Allen, Kennedy et al. - 1983
121 A VLIW architecture for a trace scheduling compiler (context) - Colwell, Nix et al. - 1987
108 Highly concurrent scalar processing (context) - Hsu, Davidson - 1986
81 HPL PlayDoh architecture specification: Version (context) - Kathail, Schlansker et al. - 1994
75 Dynamic speculation **and** synchronization of data dependences - Moshovos, Breach et al. - 1997
59 Boosting beyond static scheduling in a superscalar processor - Smith, Lam et al. - 1990
52 Dynamic memory disambiguation using the memory conflict buff.. - Gallagher, Chen et al. - 1994
38 A comparison of full **and** partial **predicated execution** suppor.. - Mahlke, Hank et al. - 1995
32 The Cydra 5 departmental supercomputer (context) - Rau, Yen et al. - 1989
31 Characterizing the impact of **predicated execution** on branch .. - Mahlke, Hank et al. - 1994
26 Guarded **execution and** branch prediction in dynamic ILP proce.. (context) - Pnevmatikatos, Sohi - 1994
24 Sentinel scheduling: A model for compiler-controlled specula.. (context) - Mahlke, Chen et al. - 1993
24 **Speculative execution** via address prediction **and** data prefet.. (context) - Gonzalez, Gonzalez - 1997
15 HP make EPIC disclosure (context) - Gwennap - 1997
15 Superscalar instruction **execution** in the 21164 Alpha micropr.. (context) - Edmondson, Rubinfeld et al. - 1995
12 The effects of **predicated execution** on branch prediction (context) - Tyson - 1994
12 Hewlett Packard Laboratories (context) - Park, Schlansker et al. - 1991
8 **Speculative execution** exception recovery using write-back su.. - Bringmann, Mahlke et al. - 1993
7 CMOS PA-RISC processor for a new family of workstations (context) - Forsyth, Mangelsdorf et al. - 1991
6 Architectural support for compile-time speculation - Smith - 1994
4 Architecture **and** Instruction Set reference manual (context) - August, Crozier et al. - 1998
3 Memory conflict buffer for achieving memory disambiguation i.. (context) - Kiyohara, Hwu et al. - 1997
1 CPU: Executing instructions in one clock cycle (context) - Crawford - 1990



The graph only includes citing articles where the year of publication is known.

Documents on the same site (http://www.crhc.uiuc.edu/IMPACT/papers/fconference.html): More
Optimization of Machine Descriptions for Efficient Use - Gyllenhaal, Hwu, Rau (1996) (Correct)
Comparing Software **and** Hardware Schemes For Reducing the.. - Hwu, Conte, Chang (1989) (Correct)
Enhanced Modulo Scheduling for Loops with Conditional.. - Warter, Haab.. (1992) (Correct)

h c e ee ec c g eg e h

Search Results

Search Results for: **[speculative compil*]**
Found **12** of **124,998 searched.**

Search within Results

GO > Advanced Search

> Search Help/Tips

Sort by: **Title** **Publication** **Publication Date** **Score** Binder

Results 1 - 12 of 12 **short listing**

1 Concert-efficient runtime support for concurrent object-oriented programming languages on stock hardware 87%
V. Karamcheti , A. Chien
Proceedings of the 1993 ACM/IEEE conference on Supercomputing December 1993

2 MaJIC: compiling MATLAB for speed and responsiveness 85%
George Almási , David Padua
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2002 Conference on Programming language design and implementation May 2002
Volume 37 Issue 5
This paper presents and evaluates techniques to improve the execution performance of MATLAB. Previous efforts concentrated on source to source translation and batch compilation; **MaJIC** provides an interactive frontend that looks like MATLAB and compiles/optimizes code behind the scenes in real time, employing a combination of just-in-time and speculative ahead-of-time compilation. Performance results show that the proper mixture of these two techniques can yield near-zero response time as ...

3 Performance monitoring: TEST: a tracer for extracting speculative threads 80%
Michael Chen , Kunle Olukotun
Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization March 2003
Thread-level speculation (TLS) allows sequential programs to be arbitrarily decomposed into threads that can be safely executed in parallel. A key challenge for TLS processors is choosing thread decompositions that speedup the program. Current techniques for identifying decompositions have practical limitations in real systems. Traditional parallelizing compilers do not work effectively on most integer programs, and software profiling slows down program execution too much for real-time analysis. ...

Retrieved 12-9-03

4 A compiler framework for speculative analysis and optimizations 80%

Jin Lin , Tong Chen , Wei-Chung Hsu , Pen-Chung Yew , Roy Dz-Ching Ju , Tin-Fook Ngai , Sun Chan

ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation May 2003

Volume 38 Issue 5

Speculative execution, such as control speculation and data speculation, is an effective way to improve program performance. Using edge/path profile information or simple heuristic rules, existing compiler frameworks can adequately incorporate and exploit control speculation. However, very little has been done so far to allow existing compiler frameworks to incorporate and exploit data speculation effectively in various program transformations beyond instruction scheduling. This paper proposes a ...

5 Using dataflow analysis techniques to reduce ownership overhead in cache coherence protocols 80%

Jonas Skeppstedt , Per Stenström

ACM Transactions on Programming Languages and Systems (TOPLAS) November 1996

Volume 18 Issue 6

In this article, we explore the potential of classical dataflow analysis techniques in removing overhead in write-invalidate cache coherence protocols for shared-memory multiprocessors. We construct the compiler algorithms with varying degree of sophistication that detect loads followed by stores to the same address. Such loads are marked and constitute a hint to the cache to obtain an exclusive copy of the block so that the subsequent store does not introduce access penalties. The simplest ...

6 The Jrpm system for dynamically parallelizing Java programs 77%

Michael K. Chen , Kunle Olukotun

ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture May 2003

Volume 31 Issue 2

We describe the Java runtime parallelizing machine (Jrpm), a complete system for parallelizing sequential programs automatically. Jrpm is based on a chip multiprocessor (CMP) with thread-level speculation (TLS) support. CMPs have low sharing and communication costs relative to traditional multiprocessors, and thread-level speculation (TLS) simplifies program parallelization by allowing us to parallelize optimistically without violating correct sequential program behavior. Using a Java virtual ma ...

7 Cool-Cache: A compiler-enabled energy efficient data caching framework for embedded/multimedia processors 77%

Osman S. Unsal , Raksit Ashok , Israel Koren , C. Mani Krishna , Csaba Andras Moritz

ACM Transactions on Embedded Computing Systems (TECS) August 2003

Volume 2 Issue 3

The unique characteristics of multimedia/embedded applications dictate media-sensitive architectural and compiler approaches to reduce the power consumption of the data cache. Our goal is exploring energy savings for embedded/multimedia workloads without sacrificing performance. Here, we present two complementary media-sensitive energy-saving techniques that leverage static information. While our first technique is applicable to existing architectures, in our second technique we adopt a more rad ...

8 Speculation: Compiler support for speculative multithreading architecture with probabilistic points-to analysis 77%

Peng-Sheng Chen , Ming-Yu Hung , Yuan-Shin Hwang , Roy Dz-Ching Ju , Jenq Kuen Lee

Proceedings of the ninth ACM SIGPLAN symposium n Principles and practice of parallel pr gramming June 2003

Speculative multithreading (SpMT) architecture can exploit thread-level parallelism that cannot be identified statically. Speedup can be obtained by speculatively executing threads in parallel that are extracted from a sequential program. However, performance degradation might happen if the threads are highly dependent, since a recovery mechanism will be activated when a speculative thread executes incorrectly and such a recovery action usually incurs a very high penalty. Therefore, it is essent ...

9 How to solve the current memory access and data transfer bottlenecks: at the processor architecture or at the compiler level 77%

Francky Catthoor , Nikil D. Dutt , Christoforos E. Kozyrakis

Proceedings of the conference on Design, automation and test in Europe January 2000

10 Integrated predicated and speculative execution in the IMPACT EPIC architecture 77%

David I. August , Daniel A. Connors , Scott A. Mahlke , John W. Sias , Kevin M. Crozier , Ben-Chung Cheng , Patrick R. Eaton , Qudus B. Olaniran , Wen-mei W. Hwu

ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture April 1998

Volume 26 Issue 3

Explicitly Parallel Instruction Computing (EPIC) architectures require the compiler to express program instruction level parallelism directly to the hardware. EPIC techniques which enable the compiler to represent control speculation, data dependence speculation, and predication have individually been shown to be very effective. However, these techniques have not been studied in combination with each other. This paper presents the IMPACT EPIC Architecture to address the issues involved in design ...

11 Evaluation of scheduling techniques on a SPARC-based VLIW testbed 77%

Seongbae Park , SangMin Shim , Soo-Mook Moon

Proceedings of the 30th annual ACM/IEEE international symposium on Microarchitecture December 1997

The performance of Very Long Instruction Word (VLIW) microprocessors depends on the close cooperation between the compiler and the architecture. This paper evaluates a set of important compilation techniques and related architectural features for VLIW machines. The evaluation is performed on a SPARC-based VLIW testbed where gcc-generated optimized SPARC code is scheduled into high-performance VLIW code. As a base scheduling compiler, we experiment with three core scheduling techniques including ...

12 Simple compiler algorithms to reduce ownership overhead in cache coherence protocols 77%

Jonas Skeppstedt , Per Stenström

Proceedings of the sixth internati nal conference on Architectural support f r programming languages and operating systems November 1994

Volume 29 , 28 Issue 11 , 5

We study in this paper the design and efficiency of compiler algorithms that remove

ownership overhead in shared-memory multiprocessors with write-invalidate protocols. These algorithms detect loads followed by stores to the same address. Such loads are marked and constitute a hint to the cache to obtain an exclusive copy of the block. We consider three algorithms where the first one focuses on load-store sequences within each basic block of code and the other two analyse the existence of l ...

Results 1 - 12 of 12 short listing